

CLAIMS

What is claimed is:

1. A method comprising:

maintaining a plurality of processes in a processing system, each with an ability to independently monitor a status of all of said plurality of processes, without the use of a master; and

enabling said plurality of processes to interact with each other to establish a priority of status, such that each of said plurality of processes can alter the priority of another of said plurality of processes without the use of a master to enable said interaction or alteration of priority.

2. A method as recited in claim 1, wherein said interaction and said alteration amongst said plurality of processes is used to enable fault tolerance for at least one of said processes in said processing system.

3. A method as recited in claim 1, wherein said status is one of: primary, to become primary, or standby.

4. A method as recited in claim 1, wherein said priority is based on a value of an identifier assigned to each of said plurality of processes.

5. A method as recited in claim 4, wherein said priority is further based on the status assigned to each of said plurality of processes.

6. A method for creating autonomy within a new process being admitted into a processing system, the method comprising:

enabling a new process to assign itself a unique, platform-independent identity, wherein an assignment of identity occurs at the time said new process is admitted into said processing system;

causing said new process to directly communicate with a plurality of process in the processing system;

enabling said new process to update a status in response to other events occurring in said processing system;

causing said new process to maintain a status in said processing system; and

causing said new process to adapt to changes in said processing system.

7. A method as recited in claim 6, wherein said status is one of: primary, to become primary, or standby.

8. A method as recited in claim 6, wherein a creation of autonomy within a new process allows for fault tolerance in a processing system without the need for a master.

9. A method for admitting a new process into a processing system, the method comprising:

admitting said new process into said processing system, such that a time at which said new process is admitted is based on whether one or more processes are being concurrently admitted with said new process;

causing said new process to broadcast a heart-beat message to notify each of a plurality of processes that said new process has been admitted into said processing system;

causing each of said plurality of processes to maintain a record identifying each of said plurality of processes; and

updating the record of each of said plurality of processes to include said new process.

10. A method as recited in claim 9, further comprising assigning said new processes a service type.

11. A method as recited in claim 9, wherein said heart-beat message broadcast by said new process includes an identifier.

12. A method as recited in claim 9, wherein said heart-beat message broadcast by said new process includes a status.

13. A method as recited in claim 9, wherein said heart-beat message broadcast by said new process includes a workload.

14. A method as recited in claim 9, wherein a maintenance of a record by each of said plurality of processes is accomplished independent of each other.

15. A method for providing fault tolerance in a processing system, the method comprising:

enabling a plurality of processes in a processing system each to broadcast a heart-beat message, wherein said heart-beat message includes an identifier for each of said plurality of processes;

enabling each of said plurality of processes to receive said heart-beat message;

causing each of said plurality of processes to maintain an individual record of said plurality of processes;

causing each of said plurality of processes to update said individual record based on said heart-beat messages;

assigning each of said processes with a status, wherein said status is one of: primary, to become primary, or standby; and

enabling said plurality of processes to negotiate a hierarchy of control amongst each other based on the broadcast and receipt of heart-beat messages by each of said plurality of processes, wherein said hierarchy of control is based on the status of each of said plurality of processes.

16. A method recited in claim 15, wherein said heart-beat message further includes a service type.

17. A method recited in claim 15, wherein said heart-beat message further includes a workload.

18. The method recited in claim 15, wherein said negotiation between said plurality of processes allow for changing a status of one or more of said plurality of processes.

19. A method for providing fault tolerance for a process within a processing system, the method comprising:

maintaining a record of a plurality of processes in said processing system, wherein said plurality of processes are each assigned an identifier;
analyzing said record of processes to determine a priority for each process;

if one of said plurality of processes has a highest priority and no other of said plurality of processes has a status of primary or has been flagged to become primary, then assigning a status of primary to said process; and

if one of said plurality of processes has a highest priority and at least one other of said plurality of processes has a status of primary or has been flagged to become primary, then assigning a status of to become primary to said process.

20. A method recited in claim 19, wherein said priority is based on a value of said identifier.

21. A method recited in claim 19, wherein assigning of a status is based on a service type of said process.

22. A processing system comprising:

a computer, wherein said computer introduces a new process into a processing system such that said new process is assigned an identifier;
means for enabling said new process to broadcast a heart-beat message to said processing system;

means for causing a plurality of processes to receive, in said processing system, said heart-beat message, wherein said heart-beat message requests said plurality of processes to individually broadcast a heart-beat message;

means for causing each of said plurality of processes to broadcast an individual heart-beat message wherein said new process receives said individual heart-beat message during a time set by a timer;

means for causing each of said plurality of processes to maintain an individual record of processes wherein said record contains an identifier, a status, a service type and a workload of each of said plurality of processes;

means for causing said plurality of processes to establish a communication amongst each other, wherein said communication provides for an establishment of priority and status for each of said processes;

means for enabling said plurality of processes to replace one or more faulty processes;
and

means for enabling said processing system to introduce one or more new processes.

23. The method recited in claim 22, wherein said broadcast of said heart-beat message includes a status.

24. The method recited in claim 22, wherein said broadcast of said heart-beat message includes an identifier.

25. The method recited in claim 22, wherein said broadcast of said heart-beat message includes a workload.

26. The method recited in claim 22, wherein said broadcast of said heart-beat message includes a service type.

27. The method recited in claim 22, wherein a replacement of one or more processes is based on the priority of each of said plurality of processes.

28. The method recited in claim 22, wherein an introduction of one or more new processes is based on a need for said new processes, wherein said need is based on a type of service.

29. A processing system comprising:
a plurality of processes, wherein each of said plurality of processes is able to independently monitor a status of each other of said plurality of processes, each of said processes communicating with each other to establish a priority of status such that at least one of said plurality of processes can change a status of each other.

30. The processing system recited by claim 29, wherein a master is not required to monitor and establish priority amongst said plurality of processes.

31. The processing system recited by claim 29, wherein said monitoring and establishment of priority amongst said plurality of processes is accomplished from the broadcasting of heart-beat messages.

32. The processing system recited by claim 29, wherein said monitoring and establishment of priority amongst said plurality of processes is used to enable fault tolerance for at least one of said processes in said processing system.

33. The processing system recited in claim 29, wherein said processing system comprises one or more computers.

34. The processing system recited in claim 29, wherein said communication between said plurality of processes does not require a master.

35. A processing system for providing fault tolerance, the processing system comprising:
- a processor; and
 - a memory containing software which, when executed by the processor, causes the processing system to perform a process comprising:
 - enabling each of a plurality of processes to transmit an individual heart-beat message wherein said heart-beat message contains an identifier of said process;
 - causing each of said processes to receive said individual heart-beat message;
 - causing each of said processes to maintain a record of said plurality of processes based on the receipt of said individual heart-beat messages;
 - causing each of said plurality of processes to update said record based on said heart-message;
 - assigning each of said processes with a status wherein said status is one of: primary, to become primary, or standby; and
 - enabling said plurality of processes to negotiate a hierarchy of control amongst each other based on the broadcast and receipt of heart-beat messages by each of said plurality of processes wherein said hierarchy of control is based on the status of each of said plurality of processes.
36. The processing system recited in claim 35, wherein said individual heart-beat message further contains a status.
37. The processing system recited in claim 35, wherein said individual heart-beat message further contains a workload.
38. The processing system recited in claim 35, wherein said individual heart-beat message further contains a service type.

39. The processing system recited in claim 35, wherein said negotiation between said plurality of processes allows for changing a status of at least one of said plurality of processes.